

散乱マップを用いた多重散乱の計算と空の色の高速表示法

東京大学大学院新領域創成科学研究科	岩 崎 慶
東京大学大学院新領域創成科学研究科	柳 田 哲 一
北海道大学大学院情報科学研究科	土 橋 宜 典
東京大学大学院新領域創成科学研究科 正会員	西 田 友 是

<あらまし> コンピュータグラフィクスによる景観表示において、空は欠かせない要素のひとつである。空の色をより写実的に表現したい場合、天空光の輝度計算を多重散乱まで考慮して行う必要があるが、この多重散乱の計算は非常に複雑であり、膨大な計算コストがかかってしまう。そこで本稿では、この複雑な計算を効率よく行うために、大気を仮想的な層(サンプリングシェル)に分割し、それらの層上での微粒子による散乱光の輝度分布を散乱マップと呼ぶテクスチャとして扱うことで、グラフィクスハードウェアを効果的に使用した手法を提案する。さらに、視点高度や太陽高度が変化した場合でも高速かつ少ない記憶量で画像を生成するため、特異値分解を用いた計算およびレンダリングの手法を提案する。

キーワード： 空の色，多重散乱，グラフィクスハードウェア，特異値分解

<Summary> The sky color is indispensable for the display of natural scenes in computer graphics. To render the photorealistic images of the sky, the calculation taking into account multiple scattering is necessary. However the calculation is so complex that enormous computational cost is required. Therefore, in this paper, we proposed a method using virtual shells (sampling shells) generated in the atmosphere. In the method, the illumination distribution of the scattering of light on the virtual shells is considered as a texture map called scattering maps. This makes it possible to calculate the sky color efficiently by using graphics hardware. Furthermore, we also propose a calculation and rendering method using the singular value decomposition to display the sky quickly with a little memory when the height of the viewpoint and the sun altitude change.

Key words: Sky Color, Multiple Scattering, Graphics Hardware, Singular Value Decomposition.

1. まえがき

自然景観の写実的な表示に関する研究はコンピュータグラフィクスの分野において非常に重要な研究分野のひとつであり、数多くの研究がなされている。本論文では、自然景観の重要な要素のひとつである空を表示するための効率的な手法を提案する。空の色は空気分子やエアロゾルなどの微粒子による光の散乱によって決まる。すな

"A Calculation of Multiple Scattering Using Scattering Maps and An Efficient Method for Rendering Sky" by Kei IWASAKI (The University of Tokyo), Tetsuichi YANAGITA (The University of Tokyo), Yoshinori DOBASHI (Hokkaido University), Tomoyuki NISHITA (Member)(The University of Tokyo).

わち、太陽光が微粒子によって散乱された光である天空光の輝度は、散乱光の輝度値を視線に沿って積分した値となる。このような光学現象をシミュレートすることは、写実的な画像を生成する上で欠かせない。つまり空を描くにあたって、多重散乱を考慮することが欠かせないといえる。多重散乱を考慮するには微粒子間での光のエネルギーの均衡を計算しなければならず、いくつかの研究がこの問題を解決するためになされてきた [1, 2, 3, 4]。しかしながらこれらの研究では写実的な空のレンダリングは可能になりながらも、多大な計算コストを必要とした。

本論文では、まず多重散乱を考慮した空の色を効率的に計算する手法を提案する。これまでの研究では、視線に沿った散乱光の積分計算を視線上に多数の仮想的なサンプル点を配置することで行っていたが、提案法ではこのサンプル点の代わりに仮想的な層を用意する。この仮想的な層のことをサンプリングシェルと呼ぶことにする。そして、このサンプリングシェルにマッピングするための散乱マップを作成する。散乱マップとはサンプリングシェル上の点における散乱光の輝度値を蓄えたもので、テクスチャの一種として扱う。これによりハードウェアを用いた高速計算が可能となる。

また本論文ではさらに、太陽高度や視点高度を変化させた際にも高速に空を描画するための手法を提案する。上述のハードウェアを用いた手法によって、視線方向を変化させた際の空の描画は実時間に行うことが可能となるが、視点高度や太陽高度の変化に対しては、実時間の描画は困難である。しかしながらフライトシミュレータのようなアプリケーションにおいては、視点高度の変化に対してモニタラティブな対応が必要となる。またそのようなアプリケーションでは地形データなどの他のデータも大量に必要であり、単純にあらゆる状況での空の色をデータとして保持しておくのはデータ量が膨大になるため適切でない。そこで本論文では、特異値分解を用いてデータ量を圧縮しながら効率的に空を描画する手法を提案する。前処理の段階において多重散乱を考慮した空の色を計算しておき、行列化する。それを、特異値分解を用いて、提案法のレンダリング手法に特化した2つのテーブルに分解することによってデータ量を削減し、視点高度や太陽高度の変化に対しても実時間で空を描画することを可能とする。

以下では、まず2章において空の表示に関する関連研究について議論する。続いて3章でサンプリングシェルと散乱マップを用いた空の色の計算法について説明し、4章では、視点高度、太陽高度が変化した場合の空の色の特異値分解を用いて高速に計算する方法を説明する。5章では提案法による結果画像を紹介し、最後に6章で本

論文をまとめ、今後の課題について述べる。

2. 関連研究

本章では空の描画に関連した研究を紹介する。Klassenは大気中の微粒子による輝度分布の変化をもとに空の色を計算する手法を提案した [5]。しかしながらこの手法では、大気を水平な層で近似したために、太陽高度が低い場合光の進路距離が無限大になってしまっていた。また大気中の空気分子の密度も二つの一定な層と仮定していた。そこで、Kanedaらは大気を球面の層とし、空気分子とエアロゾルの密度を高度に対し指数関数的に変化するものと仮定した手法を提案した [6]。さらにNishitaらはKanedaらの手法を拡張し、宇宙から見た大気の色を計算する手法を提案した [7]。Nishitaらは2次の散乱まで考慮して空の色を計算し描画する手法を提案した [8]。この手法では多重散乱を計算するのに必要な光学的距離や累算輝度を、前計算しテーブルに保存しておくことで高速な計算を可能とした。また彼らの研究によって、空の色における多重散乱の重要性も示されている。

次に、高速に空を描画する手法として、Dobashiらによって基底関数を用いて空の輝度分布を表す手法が提案された [9]。彼らはさまざまな太陽高度における空の色の輝度分布を前処理で計算し、それを基底関数で表現した。この手法では太陽高度の変化に対する高速な空の描画が可能となったが、視点高度の変化に対する対応は考慮されておらず、また基底関数は低周波領域の近似には適しているが、太陽周辺のように輝度値の大きな部分の近似には適していないといえる。Preethamらはさまざまな状況における太陽光スペクトルを近似するモデルを提案した [10]。彼らのモデルは大気のさまざまな状況下における空の色への対応を目的としたものであり、視線方向や視点高度等の変化に対応するものではなかった。最近では、Dobashiらが大気の散乱効果を高速に描画する手法を提案した [11]。この手法では散乱光の輝度値を保存したルックアップテーブルをテクスチャとして用いることでハードウェアによって高速に描画されるが、1次散乱しか考慮されていない。

本論文ではNishitaらの手法 [8] を拡張し、高速な空の描画を行う。提案法では多重散乱の計算に必要な要素を全てあらかじめ計算しテーブルに保存する。またサンプリングシェルや散乱マップの概念を導入することでハードウェアの使用に特化させ、さらに特異値分解を用いて必要なデータ量の圧縮と実時間での空の描画を可能とする。

3. 多重散乱を考慮した空の色の計算

本章では多重散乱を考慮した空の色の効率的な計算手

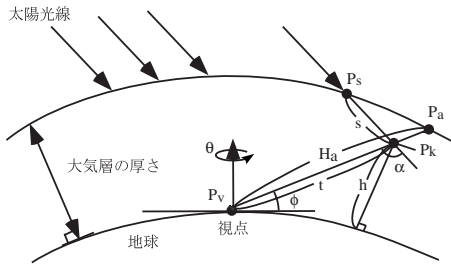


図 1 視点に到達する天空光

Fig. 1 Skylight reaching the viewpoint.

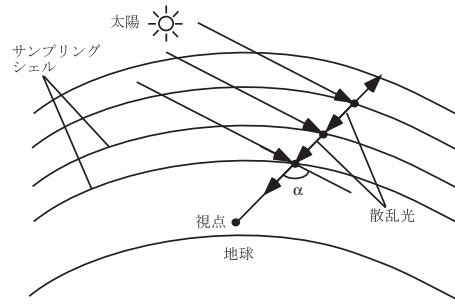


図 2 サンプリングシェル

Fig. 2 Sampling shells.

法について述べる．1,2 章で述べたように本手法ではサンプリングシェルや散乱マップといった概念を新たに提案するが，まずは天空光の計算法について述べる．そのあと提案法で用いる概念とその使用について詳解する．

3.1 散乱光の輝度計算

空の色は，太陽光の輝度分布，太陽高度，大気中の微粒子による散乱や吸収などさまざまな要素によって決まる．中でも最も重要なのは大気中の微粒子による散乱で，散乱の度合いは位相関数によって表現される．位相関数は光の波長と粒子の大きさによって決まる．大気は主に空気分子とエアロゾルの 2 つで構成されるが，空気分子による散乱はレイリー散乱と呼ばれ，エアロゾルによる散乱はミー散乱と呼ばれる．ここで 1 次散乱光を例にレイリー散乱の散乱光の計算法の説明を行う．視点 P_v に届くレイリー散乱による散乱光の輝度値 $I_v(\lambda)$ は，大気と視線の境界 P_a から P_v までの積分計算を用いて以下のように表される (図 1 参照) ．

$$I_v(\lambda) = I_{sun}(\lambda) \frac{k\beta_r(\alpha)}{\lambda^4} \int_0^{H_a} \exp(-\tau_r(s, \lambda)) \times \exp(-\tau_r(t, \lambda)) \rho_r(s) ds \quad (1)$$

ここで， λ は光の波長， $I_{sun}(\lambda)$ は太陽光の輝度， k は定数， H_a は P_a から視点 P_v までの距離とする．付録 A に示すように， $\beta_r(\alpha)$ は位相関数， ρ_r は空気分子の密度とし， $\tau_r(s, \lambda)$ は点 P_s から P_k までの光学的距離， $\tau_r(t, \lambda)$ は点 P_k から P_v までの光学的距離である．提案法では，レイリー散乱について 3 次散乱以降は無視できるため 2 次散乱まで考慮し，ミー散乱については 2 次散乱以降は無視できる大きさであるので，1 次散乱まで考慮する [8] ．

3.2 提案法の概要

視点に届く天空光の輝度は微粒子による散乱光の輝度を視線に沿って積分したものになるが，この積分計算は解析的に行うことはできない．そこで本論文ではこの計算を離散的に行うために，図 2 に示すように幾層かの仮

想的な層 (サンプリングシェル) を用意する．サンプリングシェルは，地球の中心を中心とし，大気内に設置した多層の層とする．

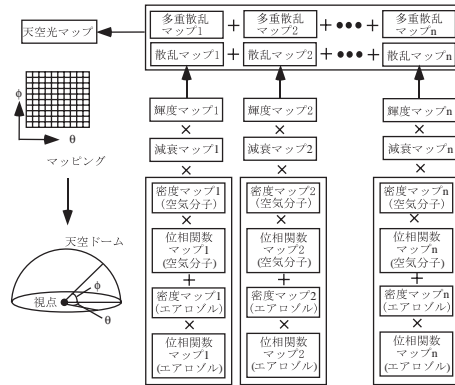


図 3 散乱マップ，天空光マップおよび天空ドーム

Fig. 3 Scattering maps, skylight map and skydome.

続いて，各サンプリングシェルにマッピングするための散乱マップを用意する．散乱マップとはサンプリングシェル上の点から視点に向かう散乱光の輝度値を保存したものである．図 3 に示す通り，最終的な画像は散乱マップに保持された輝度値を累積することで作成される．この最終的な輝度値を保存したテーブルのことを天空光マップと呼ぶことにする．加えて多重散乱も計算するために，2 次散乱光の輝度値を保持した多重散乱マップも用意する．また，各散乱マップに保存する輝度値の計算は以下のように行う．視点に届く散乱光の輝度値は 4 つの要素の積として計算できる．すなわち，サンプリングシェル上の点に到達した太陽光 (あるいは散乱光) の輝度値，その点における位相関数，微粒子の密度，その点と視点間における光の減衰率である．そこで 4 つの要素をあらかじめ計算し，保存しておくためのマップを用意する．それぞれを輝度マップ，位相関数マップ，密度マップ，減衰

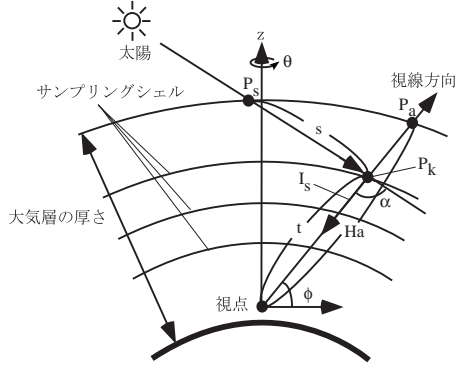


図4 サンプリングシェルによる1次散乱光の計算

Fig. 4 Calculation of single scattering of light using sampling shells.

マップと呼ぶことにする．図3に示すとおり，散乱マップはこの4つのマップを用いて計算することができる．各散乱マップを累積することによって天空光マップを作成し，天空ドームにマッピングすることによって空を描画する(図3参照)．

3.3 1次散乱光の計算

まず1次散乱を計算するための手法について述べる．図4に示すような，視点を中心， z 軸を天頂方向とした座標系を導入し，方位角を θ ，仰角を ϕ と表すことにする．このとき，視点から (θ, ϕ) 方向を見たときの天空光の輝度値 I_{sky} は以下の式で表される．

$$I_{sky}(\theta, \phi, \lambda) = I_{sun}(\lambda) \int_0^{H_a} (\beta_r(\alpha, \lambda)\rho_r(t) + \beta_m(\alpha, \lambda)\rho_m(t)) \times \exp(-(\tau_r(s, \lambda) + \tau_m(s, \lambda))) \times \exp(-(\tau_r(t, \lambda) + \tau_m(t, \lambda))) dt \quad (2)$$

ここで， ρ_r, ρ_m はそれぞれ空気分子，エアロゾルの密度， β_r, β_m はレイリー散乱，ミー散乱の位相関数， τ_r, τ_m は空気分子，エアロゾルの光学的距離とする．式(2)を計算するために，サンプリングシェルを利用する．点 P_k をサンプリングシェル上の点とすると，点 P_k における散乱光の輝度 I_s は以下の式で計算される．

$$I_s(\theta_k, \phi_k, \lambda) = I_l(\theta_k, \phi_k, \lambda)G(\theta_k, \phi_k, \lambda) \times (F_r(\theta_k, \phi_k, \lambda)D_r(\theta_k, \phi_k) + F_m(\theta_k, \phi_k, \lambda)D_m(\theta_k, \phi_k)) \quad (3)$$

ここで，各項の計算式は，

$$D_r(\theta_k, \phi_k) = \rho_r(\phi_k) \quad (4)$$

$$D_m(\theta_k, \phi_k) = \rho_m(\phi_k) \quad (5)$$

$$I_l(\theta_k, \phi_k, \lambda) = I_{sun}(\lambda) \exp(-\tau_r(s(\theta_k, \phi_k), \lambda)) - \tau_m(s(\theta_k, \phi_k), \lambda)) \quad (6)$$

$$F_r(\theta_k, \phi_k, \lambda) = \beta_r(\alpha(\theta_k, \phi_k), \lambda) \quad (7)$$

$$F_m(\theta_k, \phi_k, \lambda) = \beta_m(\alpha(\theta_k, \phi_k), \lambda) \quad (8)$$

$$G(\theta_k, \phi_k, \lambda) = \exp(-\tau_r(t(\theta_k, \phi_k), \lambda)) - \tau_m(t(\theta_k, \phi_k), \lambda)) \quad (9)$$

サンプリングシェル k 上に入射する光の輝度 $I_l(\theta_k, \phi_k, \lambda)$ を輝度マップ，サンプリングシェル上の空気分子，エアロゾルの密度 $D_r(\theta_k, \phi_k), D_m(\theta_k, \phi_k)$ を密度マップ， $F_r(\theta_k, \phi_k), F_m(\theta_k, \phi_k)$ を位相関数マップ，サンプリングシェルから視点までの減衰率 $G(\theta_k, \phi_k, \lambda)$ を減衰マップとして保存する．サンプリングシェル上の散乱光の輝度は，この4つのマップを掛け合わせることで計算される．従来法では，各視線ごとに式(2)を計算していたのに対し，提案法ではサンプリングシェル単位で，グラフィクスハードウェアによって散乱光の計算が行われるため効率的である．このように4つのマップに分解して計算することによって，太陽高度が変化した場合には式(3)の各項を計算しなおさなくとも，輝度マップおよび位相関数マップのみ計算しなおすだけでよい．

3.4 多重散乱の計算

2次散乱を計算する手法について述べる．ある方向 ω からある点 P_k に届く1次散乱光の輝度値を $I_t(P_k, \omega, \lambda)$ とすると， $I_t(P_k, \omega, \lambda)$ は方向 ω から来る1次散乱光の累積値となる(図5参照)．すなわち，ある点 P_k から視点に向かう2次散乱光の輝度値は以下の式で計算できる．

$$I_s^{(2)}(P_k, \lambda) = \frac{1}{4\pi} \int_{4\pi} \beta_r(\alpha', \lambda)\rho(\phi_k)I_t(P_k, \omega, \lambda)d\omega \quad (10)$$

この計算も解析的に行うことはできないので，点 P_k をサンプリングシェル上の点とし，点 P_k においていくつかのサンプリング方向 ω_i を用意して，離散的に式(9)の計算を行う．

$$I_s^{(2)}(P_k, \lambda) = \frac{1}{4\pi}\rho(\phi_k) \sum_{i=0}^{N_\omega} \beta_r(\alpha', \lambda)I_t(P_k, \omega_i, \lambda)\Delta\omega \quad (11)$$

ここで N_ω はサンプリング方向の数とする． $I_t(P_k, \omega, \lambda)$ も1次散乱光の各マップ同様，あらかじめ計算しておき1次散乱光輝度マップとして値を保存しておく．1次散乱光輝度マップと位相関数マップの積の和に，密度マップとサンプリングシェル k から視点までの減衰マップを

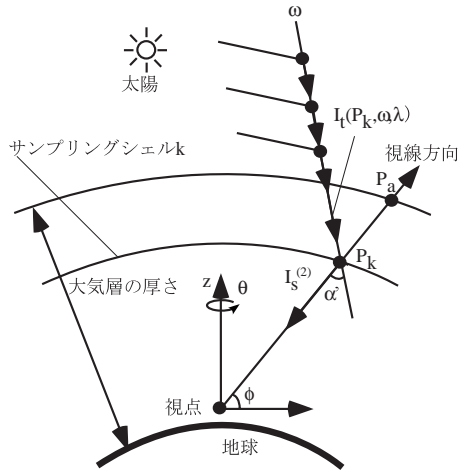


図 5 2次散乱の計算

Fig. 5 Calculation of second order scattering of light.

掛け合わせることによって、多重散乱マップ (サンプリングシェル上の2次散乱光の輝度を保持したマップ) は計算される。各シェルの多重散乱マップを累積することによって視点に到達する2次散乱光は計算される。

4. 特異値分解を用いた高速描画法

3章までに述べた方法では、従来法 [8] に比べて高速に多重散乱を考慮して空の色を計算することは可能になったが、フライトシミュレータなどの視点高度が逐次変化することが要求されるアプリケーションでは、十分な速度で空の色を計算し描画できる手法とは言いがたい。そのため、さまざまな視点高度、太陽高度における空の色をあらかじめ計算しておいて、描画時に輝度を補間する手法が考えられる。しかしながら、さまざまな方向、高度において輝度を計算する場合、膨大なデータ量を必要とする。例として、5章で述べる実験の結果得られた分割数を考える。視線方向 (θ, ϕ) を 64×64 分割し、視点高度および太陽高度をそれぞれ 32, 128 分割した場合、データ量は約 50MB ($64 \times 64 \times 128 \times 32 \times 3$ (RGB3成分) $\times 1$ (byte)) 必要となる。これは、地形データや建築物データといったさまざまな大量データを必要とするフライトシミュレータなどのアプリケーションでは問題となる。そこで、本論文では、特異値分解を利用することによってデータ量を圧縮し高速に描画する手法を提案する。

特異値分解を用いた空の描画方法は図6のような手順で行われる。提案法の描画処理は、主に前計算処理とレンダリング処理の2つから成り立っており、以下で各処理の詳細を述べる。

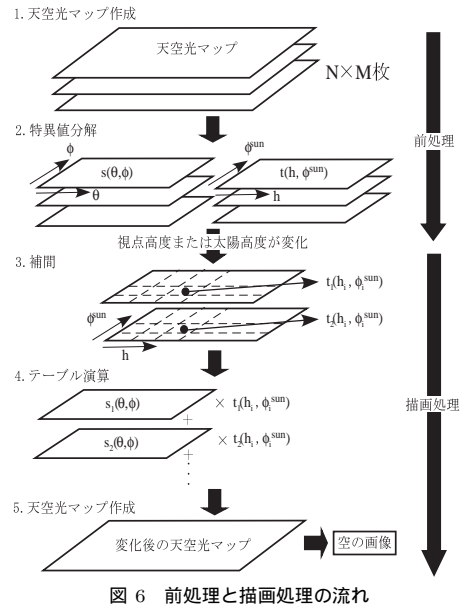


図 6 前処理と描画処理の流れ

Fig. 6 Precalculation process and rendering process.

4.1 前計算

3章で述べた散乱マップを用いて、さまざまな太陽高度および視点高度における多重散乱を考慮した空の色をあらかじめ計算する。太陽高度が ϕ^{sun} 、視点高度が h で (θ, ϕ) 方向から視点に到達する天空光の輝度を $I_{sky}(\theta, \phi, \phi^{sun}, h)$ とする。視線方向 (θ, ϕ) を $L \times L$ 分割し、太陽高度 ϕ^{sun} を M 分割、視点高度を N 分割する。

$$\theta_i = \frac{\pi i}{L} \quad (i = 0, 1, \dots, L-1) \quad (11)$$

$$\phi_i = \left(\frac{\pi}{2} + \phi(h_0)\right) \frac{i}{L} \quad (i = 0, 1, \dots, L-1) \quad (12)$$

$$\phi_i^{sun} = \frac{\pi i}{2M} \quad (i = 0, 1, \dots, M-1) \quad (13)$$

$$h_i = \frac{h_0 i}{N} \quad (i = 0, 1, \dots, N-1) \quad (14)$$

ここで、視点高度 h が上昇するにつれて水平線以下からも天空光が入射することも考慮する。提案法では、水平線から下の角度の最大値 (視点高度が最大の場合) をあらかじめ計算する。最大視点高度 h_0 の場合の水平線から下の角度を $\phi(h_0)$ とすると、 $\phi(h_0)$ は以下のような式で計算される。

$$\phi(h_0) = \arccos\left(\frac{h_0}{h_0 + H_{earth}}\right) \approx \frac{\pi}{60} \quad (15)$$

ここで、 H_{earth} は地球の半径とする。視点の最大高度 h_0 は、一般的な飛行機の最大高度である 13,000m としている。

各太陽高度，視点高度において計算した天空光の輝度値を以下のような行列 A_{sky} で表現する．

$$\begin{pmatrix} a_{(0,0,0,0)} & \cdots & a_{(0,0,M-1,N-1)} \\ \vdots & \ddots & \vdots \\ a_{(L-1,L-1,0,0)} & \cdots & a_{(L-1,L-1,M-1,N-1)} \end{pmatrix} \quad (16)$$

ここで， $a_{(i,j,k,l)} = I_{sky}(\theta_i, \phi_j, \phi_k^{sun}, h_l)$ とする．行列 A_{sky} を特異値分解することによって，天空光の輝度 $I_{sky}(\theta, \phi, \phi^{sun}, h)$ を 2 変数関数の積の和で表現することができる (付録 B 参照)．

$$I_{sky}(\theta, \phi, \phi^{sun}, h) \approx \sum_{k=1}^T \sigma_k s_k(\theta, \phi) t_k(\phi^{sun}, h) \quad (17)$$

ここで， σ_k は特異値である．本論文では 2 つの関数 $s_k(\theta, \phi)$ ， $t_k(\phi^{sun}, h)$ をテーブルとして使用している．天空光の輝度の近似の精度は，打ち切る数 T によって決まる．提案法では，太陽付近や水平付近では特異値の個数が多く必要であることから，各視線 (θ, ϕ) ごとに $T_{\theta\phi}$ を以下のように決めている．まず，散乱マップを用いて計算した天空光マップと， n 個の 2 次元テーブルによって計算した天空光マップとの輝度値について，全ての ϕ^{sun} と h における差の最大値を各視線 (θ, ϕ) ごとに計算する．その輝度値の差の最大値が閾値 ϵ 以下となる際の項の個数を $T_{\theta\phi}$ とする．提案法では，ディスプレイ上で RGB それぞれ 8 ビットで表現されることから，閾値 ϵ として $1/255$ を採用している．

4.2 描画

視点高度および太陽高度が指定された際に，2 次元テーブル $t_k(\phi^{sun}, h)$ から値を双 1 次補間により計算し，もう一つの 2 次元テーブル $s_k(\theta, \phi)$ と掛け合わせることで，視点高度および太陽高度の変化に高速に対応して空の色を計算することができる (図 6 参照)．提案法では，各視線 (θ, ϕ) で使用する特異値の数が違うため，この計算をソフトウェアで行っている．天空ドームにマッピングする際の視線方向の補間はグラフィクスハードウェアによって行っている．

5. 適用例

提案法により生成されたレンダリング結果を図 7 に示す．図 7(a) は視点高度が 7000m で太陽高度が 30 度，(b) は視点高度が 4000m で太陽高度が 5 度の空を表示した画像である．提案法によってさまざまな時刻，視点高度における写実的な空を実時間に表示することが可能となった．図 8 は 1 次散乱光のみを考慮した場合と 2 次散乱光まで考慮した場合の比較結果を表している．この図が示すとおり，2 次散乱を考慮することによってより青

みがかった空を表示することができる．図 9 はさまざまな視点高度において飛行機のcockpitからみた空を表示した画像である．図 10 は，散乱マップから計算した天空光マップ (図 10(a)) と 2 次元テーブルから計算した天空光マップ (図 10(b)) との比較を表している．これにより，両者にほぼ差が見られないことがわかる．また図 10(c) は各視線方向における近似した項の数 $T_{\theta\phi}$ を可視化したものである．近似した項の数の最小値は 1 で，最大値は 25 であった．各 RGB 波長ごとの項の数を輝度値 $(T_R(\theta, \phi)/30, T_G(\theta, \phi)/30, T_B(\theta, \phi)/30)$ で表現したものである．この図を見てわかるように，水平付近および太陽付近での多くの項が必要となることがわかる．

提案法では，実験により天空光マップの分割数 L を 64×64 とし，サンプリングセル数は 40，2 次散乱光のサンプリング数 N_ω は 15 としている．また，太陽高度の分割数 M を 128，視点高度の分割数 N を 32 としている．これは，視点高度の変化による天空光の輝度の変化に比べて太陽高度の変化による天空光の輝度変化のほうが激しいため，太陽高度の分割数が多くなっている．特異値分解により，約 50MB (4 章参照) の輝度値データを 800KB まで圧縮した．すなわち，約 1.6% のデータ量で見た目に遜色のない空の色の高速計算を可能とした．レンダリング時間は，CPU が Pentium 4 2.2GHz で GPU が GeForce4 搭載の PC で約 50fps (frames per second) であった．前処理には，輝度値の計算に 8 時間 (1 枚の天空光マップ作成に 7 秒)，特異値分解に 10 時間要した．

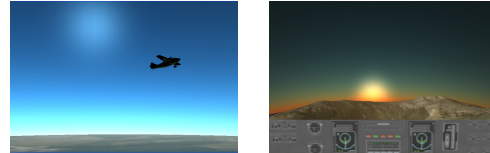
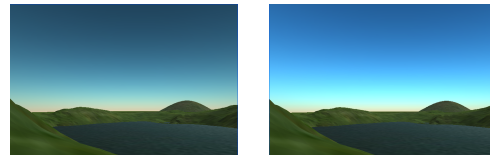


図 7 空の画像の例

Fig. 7 Examples of the sky images.



(a) 1 次散乱 (b) 2 次散乱

図 8 1 次散乱と 2 次散乱の比較

Fig. 8 Comparison of single scattering and second order scattering.



図 9 視点高度を変えた空の画像の例

Fig. 9 Examples of sky images viewed from various altitudes.

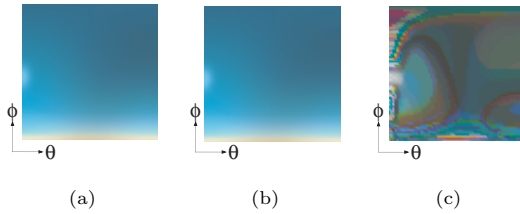


図 10 天空光マップの比較

Fig. 10 Comparison of the skylight maps.

6. おわりに

本論文では、多重散乱を考慮した空の色の効率的な計算法および高速表示法を提案した。提案法では、散乱マップとよばれるテクスチャを作成することによって、大気中の微粒子による散乱光の積分計算をグラフィクスハードウェアを使って高速に処理する。大気中にサンプリングシェルと呼ばれる仮想的な層を設定し、面上の散乱光の輝度を画像とみなすことによって高速に多重散乱を考慮した空の色を計算した。さらに、散乱マップを利用して、様々な視点高度、太陽高度における空の色をあらかじめ計算しておき、特異値分解を利用することで莫大なデータ量の4次元の輝度値のテーブルを、少ないデータ量の2次元テーブルの演算に置き換えることができた。これによって、視点高度や太陽高度が逐次変化した場合でも実時間で空の色を計算し描画することが可能となった。

今後の課題としては、前計算での特異値分解に要する計算時間の短縮および、様々な大気の状態変化にともなう空の色の変化への対応が挙げられる。

参考文献

- 1) N. Max : "Efficient light propagation for multiple anisotropic volume scattering," Proc. Eurographics Workshop on Rendering, 1994, pp.87-104.
- 2) H.W. Jensen, P.H. Christensen : "Efficient simulation of light transport in scenes with participating media using photon maps," Proc. SIGGRAPH'98, 1998, pp.311-320.
- 3) T. Nishita, Y. Dobashi, E. Nakamae : "Display of clouds taking into account multiple anisotropic scattering and sky light," Proc. SIGGRAPH'96, 1996, pp.379-386.
- 4) J. Stam, E. Fiume : "Depicting fire and other gaseous phenomenon using diffusion processes," Proc. SIGGRAPH'95, 1995, pp.129-136.
- 5) R.V. Klassen : "Modeling the effect of the atmosphere on light," ACM Trans. on Graphics, Vol.6, No.3, 1987, pp.215-227.
- 6) K. Kaneda, T. Okamoto, E. Nakamae, T. Nishita : "Photorealistic image synthesis for outdoor scenery under various conditions," The Visual Computer, Vol.7, No.5, 1991, pp.247-258.
- 7) T. Nishita, T. Shirai, K. Tadamura, E. Nakamae : "Display of the earth taking into account atmospheric scattering," Proc. SIGGRAPH'93, 1993, pp.175-182.
- 8) T. Nishita, Y. Dobashi, K. Kaneda, H. Yamashita : "Display method of the sky color taking into account multiple scattering," Proc. Pacific Graphics'96, 1996, pp.66-79.
- 9) Y. Dobashi, K. Kaneda, T. Nishita, H. Yamashita : "A fast display method of sky color using basis functions," The Journal of Visualization and Computer Animation, Vol.8, No.2, 1997, pp.115-127.
- 10) J. Preetham, P. Shirley, B. Smits : "A practical analytic model for daylight," Proc. SIGGRAPH'99, 1999, pp.91-100.
- 11) Y. Dobashi, T. Nishita, T. Yamamoto : "Interactive rendering of atmospheric scattering effects using graphics hardware," Proc. Graphics Hardware 2002, 2002, pp.99-108.
- 12) J. Kautz, M.D. McCool : "Interactive rendering with arbitrary brdfs using separable approximations," Proc. Eurographics Workshop on Rendering, 1999, pp.281-292.

A. 位相関数・光学的距離の計算式

レイリー散乱の位相関数および密度は以下の式で計算される。

$$\beta_r(\alpha) = \frac{3}{4}(1 + \cos^2 \alpha) \quad (\text{A-1})$$

$$\rho_r(s) = \exp\left(-\frac{h(s)}{H_0}\right) \quad (\text{A-2})$$

H_0 はスケールハイトと呼ばれる定数である。光学的距離 $\tau_r(s, \lambda)$ は以下の式で計算される。

$$\tau_r(s, \lambda) = \frac{4\pi k}{\lambda^4} \int_0^s \exp\left(-\frac{h(l)}{H_0}\right) dl \quad (\text{A-3})$$

エアロゾルによる散乱であるミー散乱の位相関数は

$$\beta_m(\alpha) = a(1 + 50 \cos^6 \frac{\alpha}{2}) \quad (\text{A-4})$$

で計算される。

B. 特異値分解

$K \times K$ の行列 M の特異値分解とは、以下のような行列 U, S, V の積に分解することである [12] .

$$M = USV^T \quad (\text{B-1})$$

ここで、 U と V は直交行列で、 S は対角成分が特異値 σ_k となる行列 ($\text{diag}(\sigma_k)$) とする . 行列の積 USV^T は U, V の列ベクトル \mathbf{u}, \mathbf{v} によって以下のように書き換えられる .

$$M = \sum_{k=1}^K \sigma_k \mathbf{u}_k \cdot \mathbf{v}_k \quad (\text{B-2})$$

ここで、 $\mathbf{u}_k \cdot \mathbf{v}_k$ はベクトルの外積とする . 特異値 σ_k は正の数で単調に減少するため、適切な値 $N < K$ において、行列 M を近似することができる .

$$M \approx \sum_{k=1}^N \sigma_k \mathbf{u}_k \cdot \mathbf{v}_k \quad (\text{B-3})$$

ここで、ある 2 次元関数 $f(x, y)$ が与えられ、 f の離散的な値 $f(x_1, y_1), \dots, f(x_K, y_K)$ が与えられているものとする . 離散値 $f(x_1, y_1), \dots, f(x_K, y_K)$ を以下のような行列で表現する .

$$M = \begin{pmatrix} f(x_1, y_1) & \dots & f(x_1, y_K) \\ \vdots & \ddots & \vdots \\ f(x_1, y_K) & \dots & f(x_K, y_K) \end{pmatrix} \quad (\text{B-4})$$

ここで、関数 $u_k(x), v_k(y)$ を、 $u_k(x), v_k(y)$ の離散的な値 $u_k(x_1), \dots, u_k(x_K), v_k(y_1), \dots, v_k(y_K)$ が以下の式のように、ベクトル $\mathbf{u}_k, \mathbf{v}_k$ の各要素となるような関数とする .

$$\mathbf{u}_k = (u_k(x_1), \dots, u_k(x_i), \dots, u_k(x_K))^T \quad (\text{B-5})$$

$$\mathbf{v}_k = (v_k(y_1), \dots, v_k(y_i), \dots, v_k(y_K))^T \quad (\text{B-6})$$

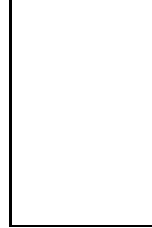
関数 f は次のような 1 次元関数 $u_k(x), v_k(y)$ の積の和で近似することができる .

$$f(x, y) \approx \sum_{k=1}^N \sigma_k u_k(x) v_k(y) \quad (\text{B-7})$$

これを応用することによって、提案法では天空光の輝度 $I_{sky}(\theta, \phi, \phi^{sun}, h)$ を 2 次元関数 $s_k(\theta, \phi), t_k(\phi^{sun}, h)$ に分解している .

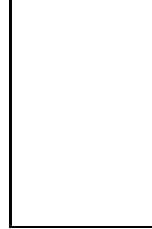
(2004 年 5 月 24 日受付)

岩崎 慶



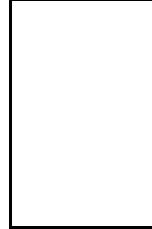
平 11, 東京大学理学部卒 . 平 16, 同大学院新領域創成科学研究科博士後期課程修了 . 科学博士 . 現在, 和歌山大学システム工学部助手 . 主として, コンピュータグラフィックス, 特に自然物の表示に関する研究に従事 . 情報処理学会会員 .

柳田 哲一

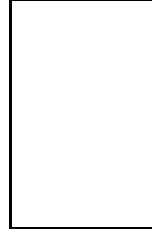


平 13, 東京大学理学部卒 . 平 16, 同大学院新領域創成科学研究科博士前期課程修了 . 同年, 株式会社ピクセラに入社 .

土橋 宜典



西田 友是 (正会員)



昭 46, 広島大学工学部卒業 . 昭 48, 同大学院工学研究科修了 . 同年, マツダ に入社 . 昭 54 年, 福山大学工学部講師 . 昭 59, 同助教授 . 平 2, 同教授 . 平成 10, 東京大学理学部教授 . 平 11, 同大学院新領域創成科学研究科教授となり, 現在に至る . コンピュータグラフィックスの研究に従事 . 工学博士 . 情報処理学会, 電子情報通信学会, 画像電子学会, ACM, IEEE 各会員 .